

Enhancing Guideline-based decision support with distributed computation through local mobile application

Erez Shalom , Yuval Shahar , Ayelet Goldstein , Elinor Ariel , Silvana Quaglini , Lucia Sacchi , Nick Fung , Valerie Jones , Tom Broens , Gema García-Sáez , and Elena Hernando

Abstract. We introduce the need for a distributed guideline-based decision support (DSS) process, describe its characteristics, and explain how we implemented this process within the European Union's MobiGuide project. In particular, we have developed a mechanism of sequential, piecemeal *projection*, i.e., 'downloading' small portions of the guideline from the central DSS server, to the local DSS in the patient's mobile device, which then applies that portion, using the mobile device's local resources. The mobile device sends a *callback* to the central DSS when it encounters a triggering pattern predefined in the projected module, which leads to an appropriate predefined action by the central DSS, including sending a new projected module, or directly controlling the rest of the workflow. We suggest that such a distributed architecture that explicitly defines a dialog between a central DSS server and a local DSS module, better balances the computational load and exploits the relative advantages of the central server and of the local mobile device.

Keywords: clinical guidelines, decision support, distributed computing

1 Introduction – the need for distributed decision support

Traditional guideline-based decision-support systems (DSS) mainly target physicians at the point of care. Physicians, however, are not the

only potential recipients of advice; patients can also be empowered to manage their own disease, especially through the use of applications running on personal mobile devices. Nevertheless, mobile-based applications might miss critical recommendations based on clinical guidelines [1], if they are not connected to a guideline server.

Thus, the main goal of the MobiGuide project [2] is to develop a distributed patient guidance system that integrates hospital and monitoring data into a Personal Health Record (PHR) accessible by patients and physicians and that provides personalized, secure, clinical-guideline-based guidance both inside and outside standard clinical environments. The distributed model of such a framework might be implemented as service oriented architecture [3], which might be more suitable for distributing a process inside a hospital. However, in the case of the MobiGuide project, we have chosen to split the architecture into two main components: a back-end DSS (BE-DSS) residing on a server system (this could be a cloud server, or, as in our case, on-premise servers in hospitals), and a mobile DSS (mDSS) residing on the patient's mobile device. The local mDSS components are necessary to distribute computationally intensive monitoring and decision-making processes, with respect to data and knowledge requirements, at the local device level.

1.1 The factors for decision-support distribution

Consider a process of monitoring each patient's high-frequency ECG sensor signals by means of patient-worn biosensors, linked via blue-tooth to the mobile device, and detecting a pattern of Atrial Fibrillation (AF) that can be abstracted from these signals by the mDSS, and sent to the central PHR to support a guideline-based recommendation to the patient and/or physician by the BE-DSS: such a distribution of labor, in which the AF detection for each patient is done by the mDSS is natural, and prevents an over burdening of the central BE-DSS server. Furthermore, the local mDSS is also essential for continuity of care when for some reason there is no internet connection to the central DSS; we would still want the patient to be provided with alerts relevant to the latest guideline by which she was being managed. However, not all decisions can be made locally on the mobile (despite ever increasing processing power and storage). Some decisions require the full historical (longitudinal) patient data, which should not reside on the mobile device. In some cases the decision to be made is a part of a long-term

plan in the complete clinical guideline knowledge base (which is continuously maintained by medical domain experts) and in other cases broader medical declarative (interpretation) knowledge may be needed in order to switch to another branch in the guideline, or even to a completely different guideline; such knowledge resides only on the central knowledge-base server. In such cases, we want the mDSS, when encountering certain local predefined (temporal) patterns, to send a *callback* message to the BE-DSS, asking for further instructions, resulting in a BE-DSS recommendation, and even a completely new projected guideline or guideline branch.

We adopted a distribution policy based on a number of factors in order to determine whether various decisions and actions should be applied at the BE-DSS level or at the mDSS level. These factors are: who is the actor of the decision (patient or physician); the temporal horizon of future recommendations e.g. immediate alerts by the mobile device when some value is out of range, versus longer-term guideline-based decisions at the server); the data and knowledge resources needed for the decision; the need for PHR access, and a consideration of where a potential personalization of the guideline should reside. These principles need to be considered by the knowledge engineer and expert physicians during the knowledge specification phase. However, once it is decided that the decision can be applied by the mDSS, i.e., can support the patient when he/she is not with the physician, the mDSS needs the relevant guideline knowledge. Therefore, there is a need to *project* ('download') small portions of the guideline, referred as *projections*, from the BE-DSS to the mDSS, and applies it using the mobile's local resources.

2 The knowledge projection model methodology

As the projection relates to *knowledge* and not to data, we refer to it as a *knowledge projection*, or *projection* for short. Each portion of the guideline which can be identified as a self-contained executable knowledge package to be potentially projected and applied in the mDSS, is called a *projection*. It is identified (tagged) as a *projection point* ("projection point" is a point in the guideline in which the designer made the decision to project a part of the guideline to the mobile device.). Only parts of the guideline that are applicable to the current state of the patient are projected. Hence, the projected knowledge includes implicitly also knowledge about the current state of the patient. The

main challenge in designing the projection process during the guideline specification time is to choose at which level (mDSS or BE-DSS) the plans and decisions should be placed, and which patterns should trigger callbacks to the central server. When choosing these projection-points in the guideline, consideration should be given to not only to technical analysis methodologies, but also to clinical properties, such as whether the decision could be taken by the patient alone, supported by the mobile device (in that case, it could be delegated to the mDSS) or whether it should be taken by the physician using the patient's full historical record and the complete guideline knowledge (should be performed on the BE-DSS). To facilitate the process of identifying projection points, we enhanced our current guideline-specification methodology: The first phase in this methodology is to create a local consensus of the guideline. The local consensus is a structured document that describes schematically the interpretation of the guideline agreed upon by both the expert physicians and the knowledge engineers, and includes the clinical directives of the guideline and the semantic logic of the specification language [4]. After creating the local consensus, the elicitation phase splits into two parallel branches: The first branch is the “traditional” one and is directed at the care professional; the second branch includes conceptually a parallel part of the process that focuses on the patient's behavior and is added to the mobile-customized guideline. Thus, we refer to the resulting specification as the *Parallel Workflow* [5]. Each parallel workflow includes parts of the guideline that might be potentially projected. After this process, the knowledge engineer might use the characteristics listed in the previous section to determine the projection points. Currently, this process is done manually by setting as "True" a special property called "isProjected" that we added to the knowledge schema. However, based on our experience, we have started the process of designing an automatic broker that will suggest a set of projection points within the guideline, during the specification, or following phase.

Figure 1 shows an example of the projection workflow: After the physician starts the Gestational Diabetes Mellitus (GDM) guideline, the plan "monitoring blood pressure twice a week", previously tagged as a projection point is downloaded to the mDSS and applied by the mobile device [6]. When the mDSS encounters the triggering pattern "abnormal blood pressure", a callback message is sent to BE-DSS, which reacts with a new projection "monitoring the blood pressure daily". Via

the back-end, the physician can inspect and explore data collected from the patient mobile, or respond to recommendations.

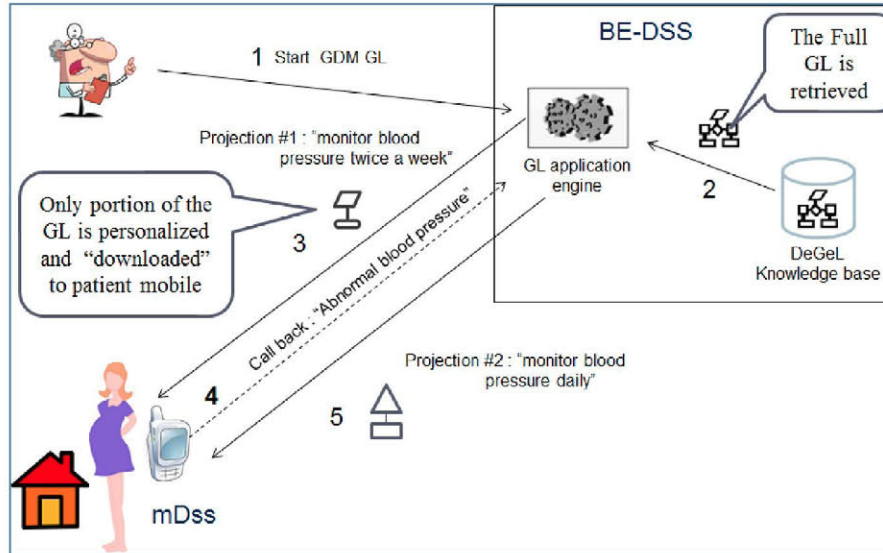


Figure 1. An example of a projection workflow for the "monitoring blood pressure" plan, and the tagging of the "blood pressure" plan as a projection.

3 The projection schema and support of personalization

There are three principles that we suggest should be followed when defining the projection schema:

1. *Separation of declarative and procedural knowledge*: One of the principles of defining the projection schema was to benefit from using the DeGeL [7] knowledge schema which supports the separation between declarative and procedural knowledge.. Therefore, the knowledge projection model is separated into two types: (1) a Declarative projection model – including concepts (raw data and simple abstractions), and personal (patient-specific) events that induce predefined customized contexts in the guideline, within which the guideline's actions might be modified (e.g., a personal *Wedding* event might induce for a particular patient the predefined "*high carbohydrate intake*" customized context mentioned in the guideline), and (2) a Procedural projection model– including plans for general treatments, or for specific personal contexts treatments.

2. *Support of Personalization*: The personal events are sent to the mDSS before starting the guideline application session because the mDSS has to initialize the patient (mobile) interface. For example, the mDSS sends to the patient interface the list of personal events that were selected during the initial registration session as inducing certain predefined contexts in the customized guideline. These personal events will be shown to the patient at the start of the application so that he or she can choose the current personal event (e.g. "holiday", "work", "regular" or "wedding"). For example, after choosing the personal event "work" and starting the guideline application, when setting a new personal event, e.g., "holiday", a second additional procedural projection might be sent to the mDSS with different monitoring scheduling.

3. *Definition of appropriate callbacks* – When a certain predefined pattern is detected at the mobile-device end by the mDSS, it triggers the mDSS to send a message to the BE-DSS to "take control" and continue the application of the GL. One example is a pattern of several consecutive days of high fasting blood glucose levels, signifying that the patient is not well controlled. This message is called a "callback", and is predefined in the projection. A callback might lead to the sending of a new projection to the mobile. Thus, the projections and callbacks support a continuous dialog between the BE-DSS and the mDSS. In any case, the BE-DSS continues to apply the GL and to send the appropriate personalized projections to the mDSS when necessary.

4. Discussion

The projection and callback mechanism that we are implementing supports a continuous dialog between the BE-DSS and the mDSS, and exploits the relative advantages of the different computational architectures and their respective access to clinical data and medical knowledge. We believe that the principles underlying this two-tiered architecture are rather general and support both functional (e.g., scalability) and non-functional (e.g., efficiency, security) requirements.

Initially, the architecture included decision support (to both patients and clinicians) provided only by the BE-DSS. That, however, did not easily cater to multiple local monitoring actions and reminders or to the robustness of the system with respect to connectivity; thus, we added the projection and callback mechanisms, splitting the decision-support

tasks between the BE-DSS and the mDSS. Currently, we are in year 3 of a 4-year project. We have implemented the distributed projection process in the case of the GDM and AF guidelines: In the case of the GDM guideline, 39 projection points were tagged, and in the case of the AF guideline, 20. Most of the projection points were cyclical plans for measurements (e.g. blood glucose), and callback messages which can significantly reduce the computational load on the BE-DSS. In the coming year, we will perform a pilot study to test the feasibility of using a distributed DSS architecture to manage patients managed under one of two guidelines: The GDM guideline in the case of the Sabadell Hospital in Barcelona, Spain, and the AF guideline in the case of the Fondazione Salvatore Maugeri, Pavia, Italy.

Acknowledgements. The MobiGuide project (<http://www.mobiguide-project.eu/>) has received funding from the EU's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 287811

References

1. Chomutare T, Fernandez-Luque L, Arsand E, Hartvigsen G. Features of mobile diabetes applications: review of the literature and analysis of current applications compared against evidence-based guidelines. *J Med Internet Res*. 2011;13(3):e65. doi: 10.2196/jmir.1874
2. Peleg, M., Shahr, Y., Quaglini, S. Making healthcare more accessible, better, faster, and cheaper: the MobiGuide Project. *European Journal of ePractice: Issue on Mobile eHealth* 2014; 20, pp. 5-20
3. Besana P, Barker A, Towards Decentralised Clinical Decision Support Systems. *Advanced Computational Intelligence Paradigms in Healthcare 5, Studies in Computational Intelligence Volume 326*, 2011, pp 27-44
4. Shalom, E., Shahr, Y., Taieb-Maimon, M., Lunenfeld, E., Bar, G., Yarkoni, A., Young, O., Martins S.B., Vaszar, L.T., Goldstein, M.K., Liel, Y., Leibowitz, A., Marom, T., and Lunenfeld, E. (2008). A quantitative evaluation of a methodology for collaborative specification of clinical guidelines at multiple representation levels. *The Journal of BioMedical Informatics* 41(6), 889-903.
5. Sacchi, L., Fux, A., Napolitano, C., Panzarasaa, S., Peleg, M., Quaglini, S., Shalom, E., Soffer, P., Tormene, P. (2013). Patient-tailored Workflow Patterns from Clinical Practice Guidelines Recommendations. *Medinfo* 2013
6. García-Sáez, G., Rigla, M., Martínez-Sarriegui, I., Shalom, E., Peleg, M., Broens, T., Pons, B., Caballero-Ruiz, E., Gómez, E.J., and Hernando, M.E. "Patient-oriented Computerized Clinical Guidelines for Mobile Decision Support in Gestational Diabetes" *Journal of Diabetes Science Technology*, 2014 DOI: 10.1177/1932296814526492.
7. Shahr, Y., Young, O., Shalom, E., Galperin, M., Mayaffit, A., Moskovitch, R., et al. A framework for a distributed, hybrid, multiple-ontology clinical-guideline library, and automated guideline-support tools. *J Biomed Inform* 2004;37(5):325-344.